Linux System Architecture

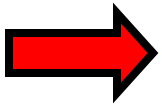
By : Amir Hossein Payberah

payberah@yahoo.com

Tehran Polytechnic University

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

Contents



- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

Tehran Polytechnic University

What is Kernel ?

- Modules or sub-systems that provide the operating system functions.
- The Core of OS
- It is written in C

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

Tehran Polytechnic University

Kernel Architecture Overview

- User Space
- Kernel Space
- Data Flow Between User Space and Kernel Space

Tehran Polytechnic University

User Space

- The User Space is the space in memory where user processes run.
- This memory is above the Kernel.
  - It includes the rest of available memory.
- This Space is protected.
  - The system prevents one process from interfering with another process.
  - Only Kernel processes can access a user process

Kernel Space

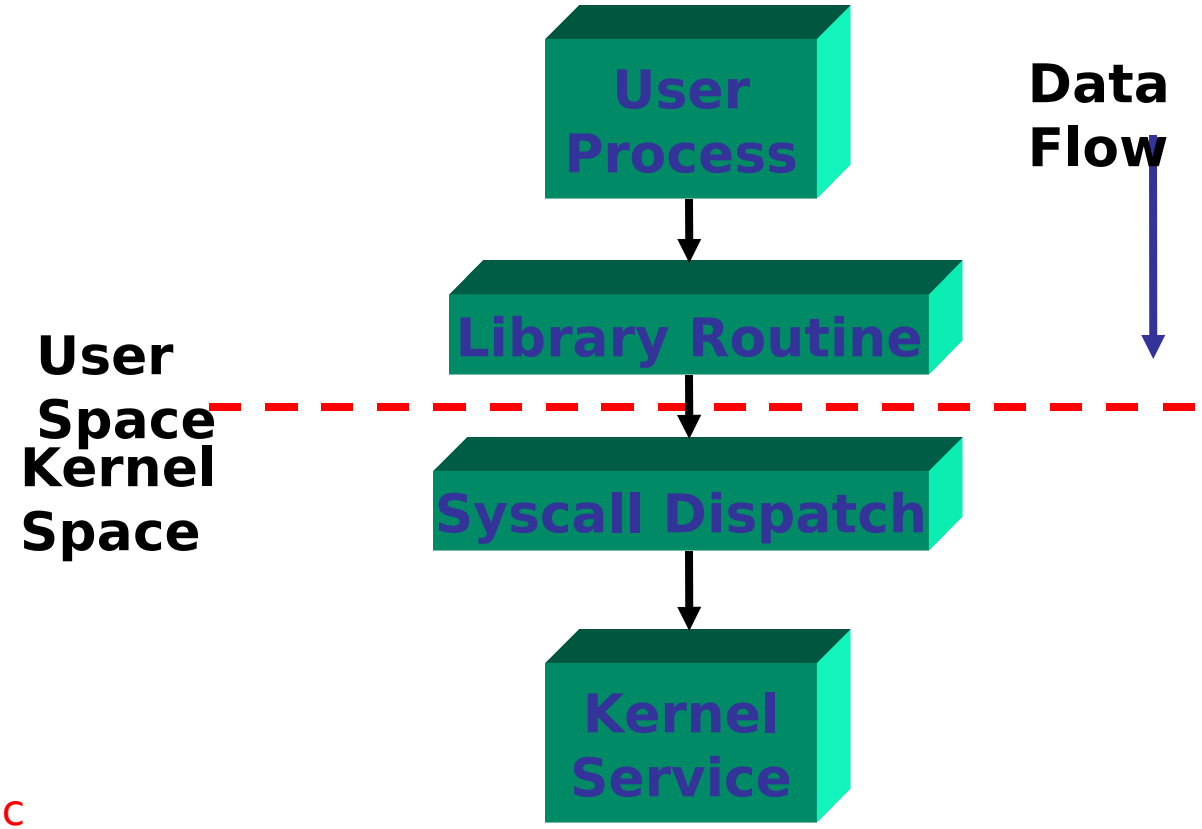
- The Kernel Space is the space in memory where all kernel services are provided via kernel processes.
- The user has access to it only through the system call.
  - A user process becomes a kernel process when it executes a system call.

System Call

- User Space and Kernel Space are in different spaces.
- When a System Call is executed, the arguments to the call are passed from User Space to Kernel Space.

Tehran Polytechnic University

User Space and Kernel Space Relationship



Tehran Polytechnic University

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

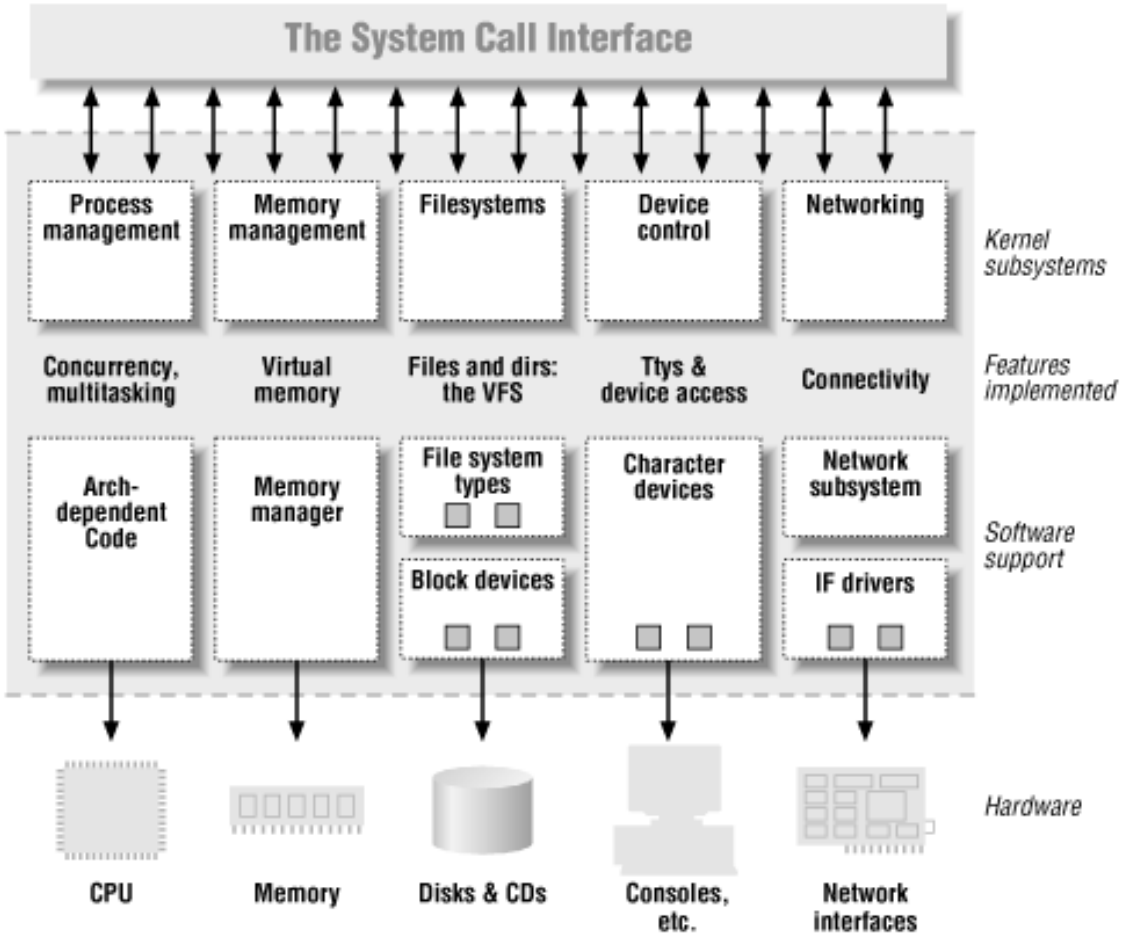
Tehran Polytechnic University

Kernel Functional Overview

- File System
- Process Management
- Device Driver
- Memory Management
- Networking

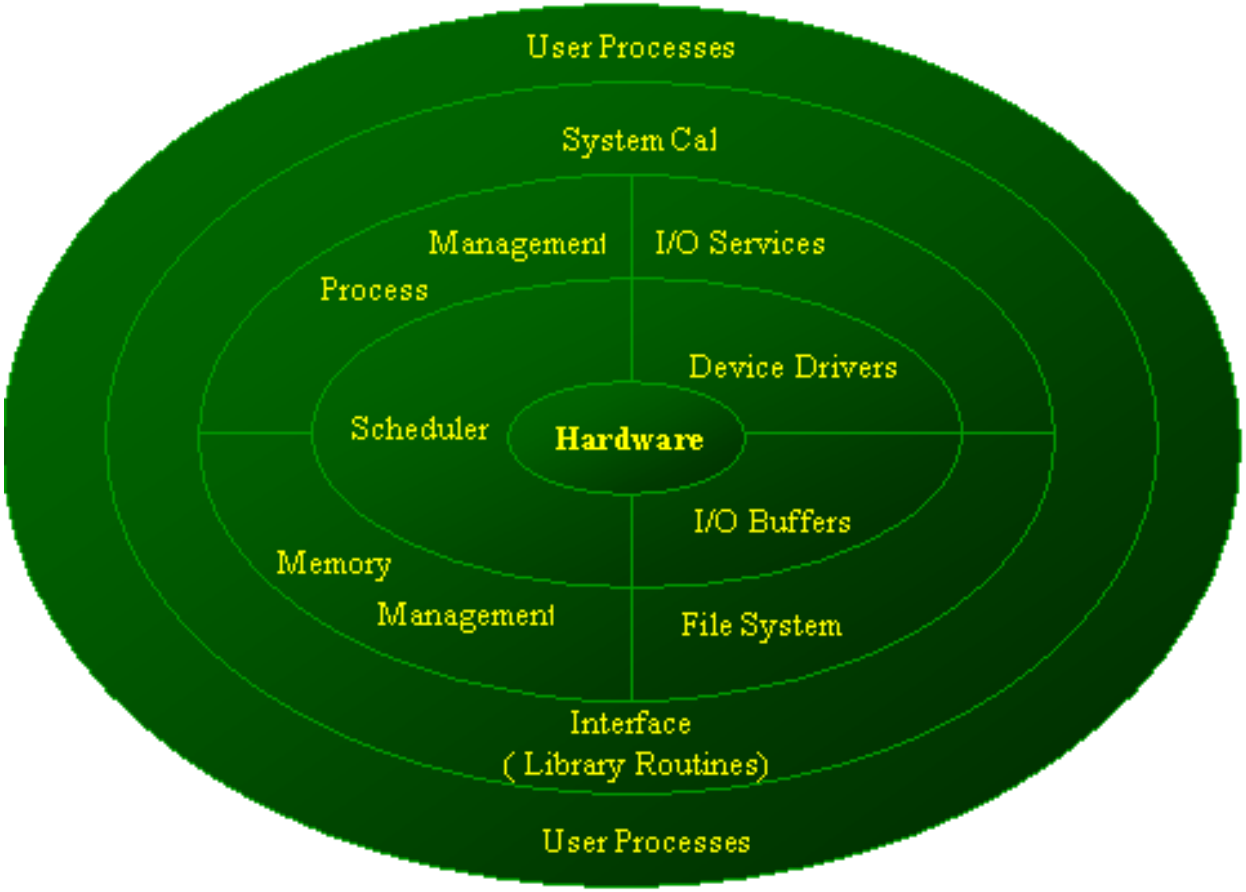
Tehran Polytechnic University

Kernel Functional Overview



Tehran Polytechnic University

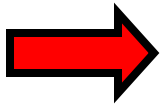
Functional Layer & Architectural Layer



Tehran Polytechnic University

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking



Tehran Polytechnic University

File System

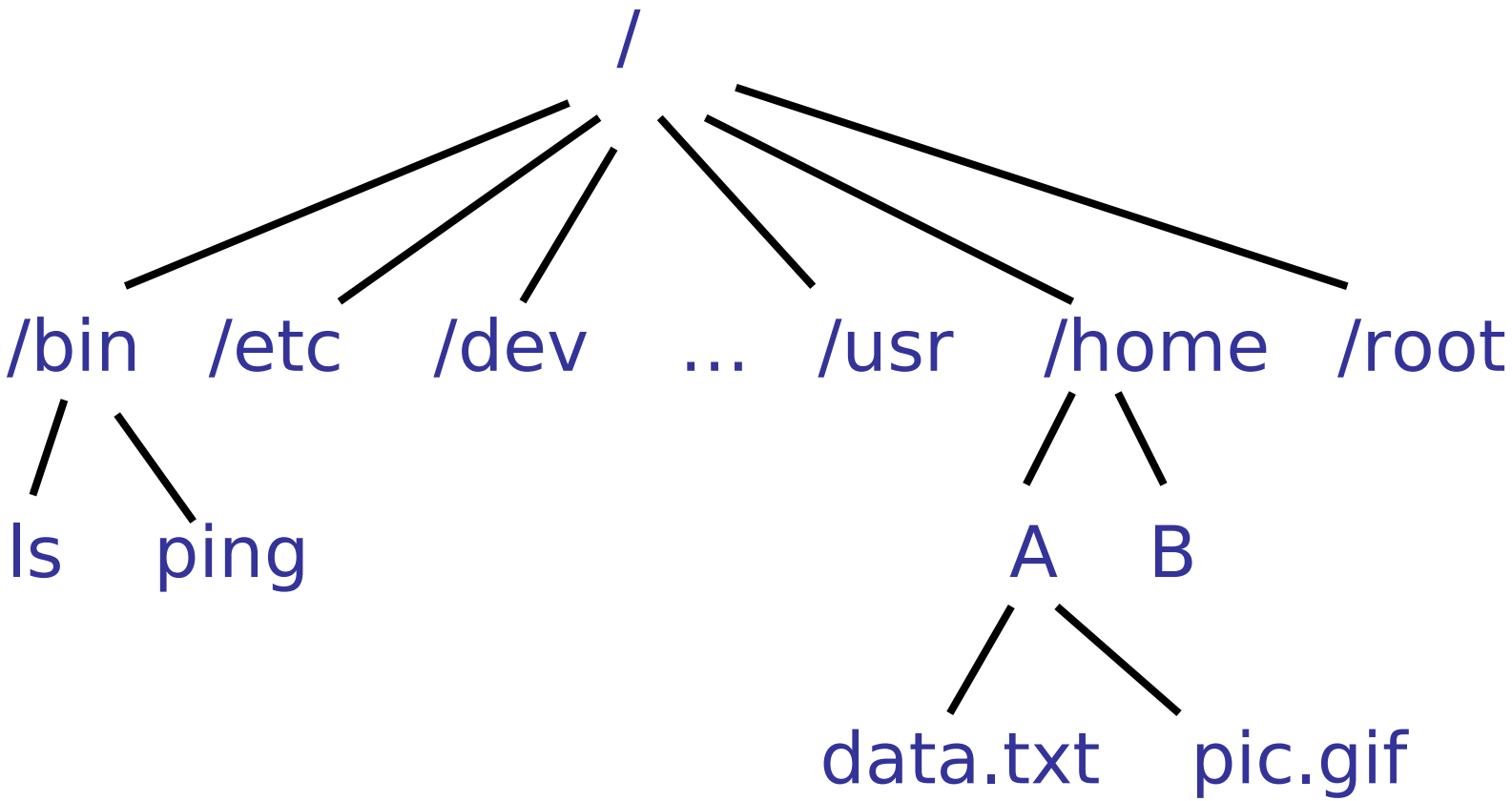
- It is responsible for storing information on disk and retrieving and updating this information.
- The File System is accessed through system calls such as : open, read, write, ...
- Example :
  - FAT16, FAT32, NTFS
  - ext2, ext3
  - ...

Tehran Polytechnic University

File System (Cont.)

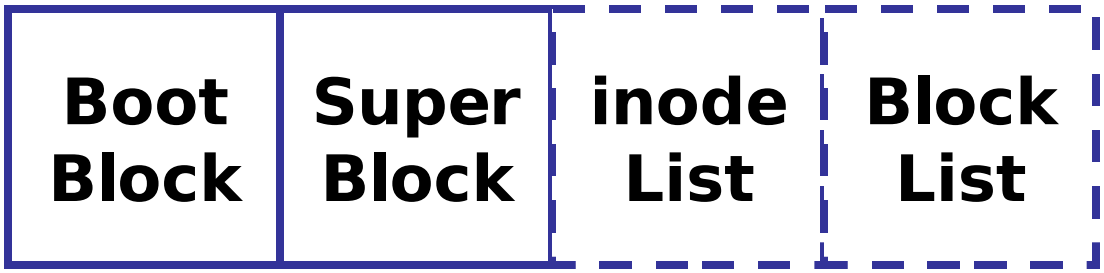
- The Unix system has the following types of files:
  - Ordinary Files
    - Contain information entered into them by a user, an application or …
  - Directory Files
    - Manage the cataloging of the file system
  - Special Files (devices)
    - Used to access the peripheral devices
  - FIFO Files for Pipes

File System (Cont.)



Tehran Polytechnic University

File System Structure



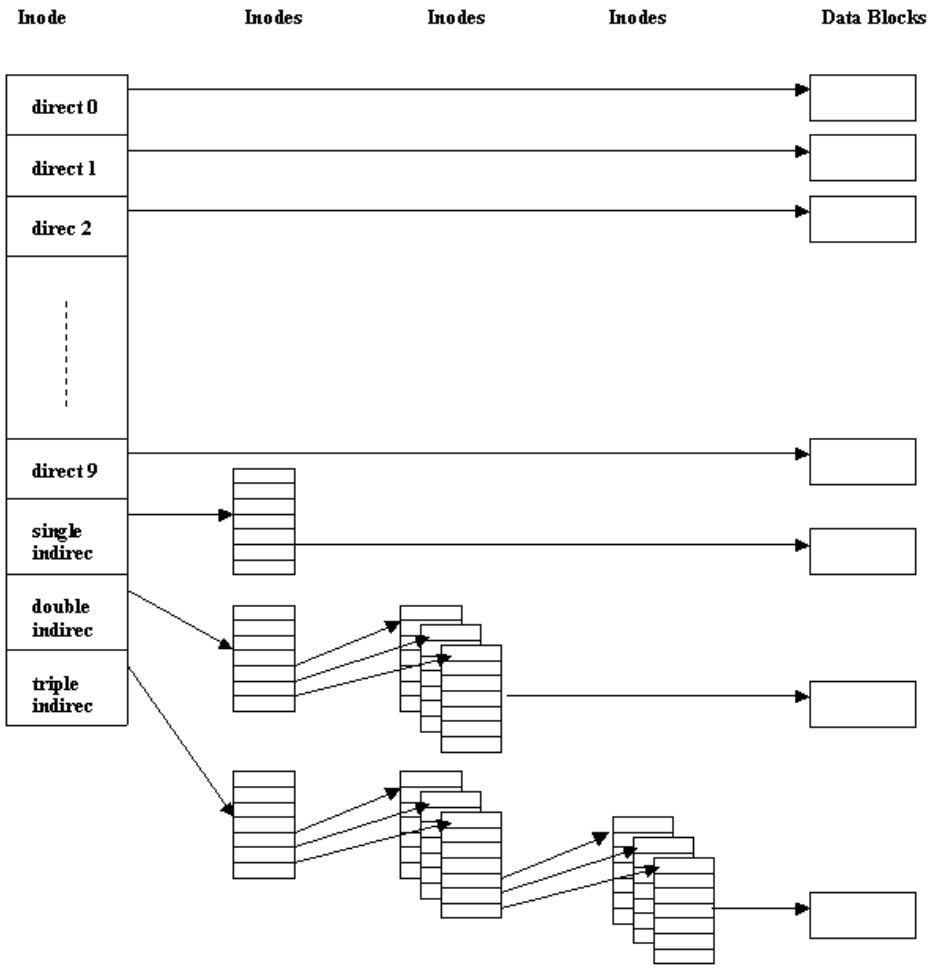
- Boot Block : information needs to boot the system
- Super Block : File System Specifications
  - Size
  - Max. number of files
  - Free blocks
  - Free inodes
- inode List
- Block List : The files data

Tehran Polytechnic University

Inode

- Each file has an inode structure that is identified by an i-number.
- The inode contains the information required to access the file.
- It doesn’t contain file name.

Inode (Cont.)



Tehran Polytechnic University

Directories

File Name	inode Number

Tehran Polytechnic University

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

Tehran Polytechnic University

Process Management

- The Unix OS is a time-sharing system.
- Every process is scheduled to run for a period of time (time slice).
- Kernel creates, manages and deletes the processes

Process Management (Cont.)

- In the Unix system process 0, called the swapper, is always assigned to the process and CPU scheduler.
- It manages the complete operation of process scheduling and swapping.
- Process 0 is create as a part of system boot-up.
- Every other process in the system is create as the result of a fork system call.
- The fork system call splits a process into two processes (Parent and Child).
- Each process has a unique identifier (Process ID).

Process Management (cont.)

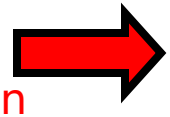
- Each process is represented by a task_struct data structure.
  - It contains the specifications of each process such as:
    - State
    - Scheduling information
    - Identifier
    - ...
- The task_vector is an array of pointers to every task_struct data structure in the system.
  - This means that the maximum number of processes in the system is limited by the size of the task vector

Type of Processes

- Running
  - The process is either running or it is ready to run.
- Waiting
  - The process is waiting for an event or for a resource.
- Stopped
  - The process has been stopped, usually by receiving a signal.
- Zombie
  - This is a halted process which, for some reason, still has a task_struct data structure in the task vector.

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking



Tehran Polytechnic University

Device Driver

- Associated with each physical device or virtual device is a piece of code, called device driver, which manage the device hardware.
- The main functions of device driver:
  - Setting up hardware on initialization.
  - Bringing the associated devices into and out of services.
  - Receiving data from the hardware and passing it back to the kernel.
  - Sending data from the kernel to the device.
  - Detecting and handling device errors.

Type of devices

- Every thing in Unix is a file.
  - Each device is like a file.
- Character devices
  - A character (char) device is one that can be accessed as a stream of bytes.
  - Example : Keyboard, Mouse, ...
- Block devices
  - A block device can be accessed only as multiples of a block, where a block is usually one kilobyte of data or another power of 2.
  - Example : disk, ...

Major Number and Minor Number

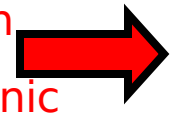
- Major Number
  - The major number identifies the driver associated with the device.
- Minor Number
  - The minor number is used only by the driver specified by the major number; other parts of the kernel don't use it.
  - It is common for a driver to control several devices, the minor number provides a way for the driver to differentiate among them.

Device Driver (Cont.)

```
crw-rw-rw- 1 root   root     1, 3   Feb 23 1999  null
crw------- 1 root   root    10, 1   Feb 23 1999  psaux
crw------- 1 rubini tty      4, 1   Aug 16 22:22 tty1
crw-rw-rw- 1 root   dialout  4, 64  Jun 30 11:19 ttyS0
crw-rw-rw- 1 root   dialout  4, 65  Aug 16 00:00 ttyS1
crw------- 1 root   sys      7, 1   Feb 23 1999  vcs1
crw------- 1 root   sys      7, 129 Feb 23 1999  vcsa1
crw-rw-rw- 1 root   root     1, 5   Feb 23 1999  zero
```

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking



Tehran Polytechnic University

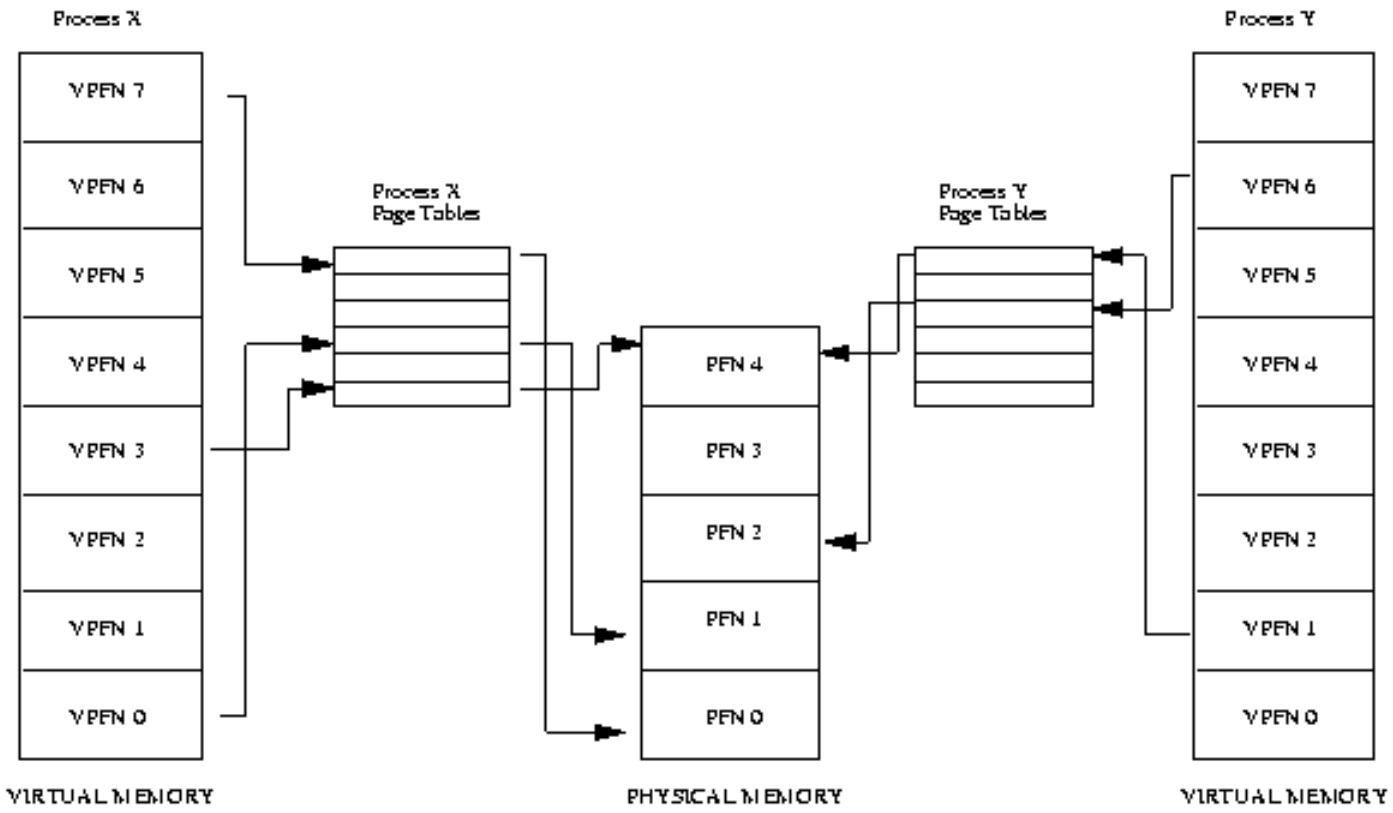
Memory Management

- Physical memory is divided into segments of equal size, called pages.
- Type of memory:
  - Physical memory
  - Virtual memory
  - Swap memory

Virtual and swap memory

- Swap
  - It is a configurable partition on disk treated in a manner similar to memory.
- Virtual memory
  - Processes swapping in and out of memory on a recurring basis fragment physical memory.
  - It is impossible to guarantee that exactly the same physical memory blocks will be available to a specific process.
  - Virtual address space solves this problem by assigning contiguous virtual memory blocks to a process.
  - These blocks are mapped to physical memory when a process is swap in, and to swap device memory when the process is swapped out.

Physical and Virtual memory



Tehran Polytechnic University

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

Tehran Polytechnic University

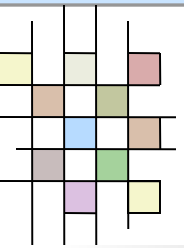
Networking

- The first integrated communication capability in Unix was developed for Berkeley Unix 4.2bsd as the sockets implementation.
- Sockets provide a programming interface for networking.

Contents

- What is Kernel?
- Kernel Architecture Overview
  - User Space
  - Kernel Space
- Kernel Functional Overview
  - File System
  - Process Management
  - Device Driver
  - Memory Management
  - Networking

Tehran Polytechnic University



Question?

Tehran Polytechnic University